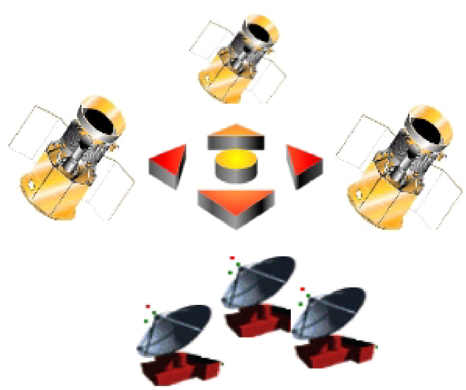
AEROFLEX ALTAIR CYBERNETICS
4201 Northview Drive, Bowie, Maryland 20716
(800) 7ALTAIR

ARCHITECTURE DESCRIPTION

for the

Onboard Automation for Autonomous Constellation Operation



Delivered under Contract Number: S-49965-G

Revision 1.5 – Initial Release

Provided to:
Goddard Space Flight Center
Attn: Julie Breed
Greenbelt, MD 20771-0001

TABLE OF CONTENTS

1 SCOPE

This section provides an architecture description for the onboard automation for autonomous constellation operation.

1.1 Identification

This architecture description document applies to an effort by Aeroflex Altair Cybernetics Corporation (Altair) to demonstrate functionality of onboard automation and operation of an autonomous homogenous constellation. It is a detailed description and explanation of the *"Concept of Operations for the Onboard Automation for Autonomous Constellation Operations"* document.

1.2 System Overview

The focus of this effort is to demonstrate the functional capabilities of using the ***Altairis Mission Control System (MCS)*** ***TM*** as an on-board constellation control system and to demonstrate system architecture for autonomous constellation control. This architecture description applies to a demonstration of autonomous homogeneous constellation and ground control system. Three virtual spacecraft will be created using ***Altairis MCS™***. The spacecraft subsystems will be modeled and simulated using Finite State Modeling. The spacecraft and the ground station will be linked as nodes, where spacecraft to spacecraft and spacecraft to Mission Operations Center (MOC) communication is required. The ***Altairis MCS™*** will also be used to simulate the functionality of a MOC. The communication bandwidth and nodal connections will be monitored as the spacecraft and the constellation configuration changes. This document describes both the operational architecture, and the hardware and software architecture.

This demonstration will only be able to demonstrate a selected subset of the domains and services available. This will be a functional demonstration based on COTS hardware for ground-based use. No space-qualified hardware will be used at this time. Follow-up projects will migrate this prototype to space qualified hardware and eventually to on-orbit test and verification.

Due to hardware availability, the constellation demonstrated consists of three satellites and a ground station. This should be sufficient to demonstrate the concepts of constellation operations, but future work should expand the number of nodes to determine any additional issues that arise from a large constellation.

2 REFERENCED DOCUMENTS

This section provides a list of reference documents.

1. Statement Of Work (SOW) for NASA Goddard Space Flight Center. S-49965-G
2. Aeroflex-Altair Cybernetics "*Concept of Operations for the Onboard Automation for Autonomous Constellation Operations*". Presented to Julie Breed GSFC Code 588, June 2001.
3. Stephen J. Talabac, *Spacecraft Constellations: The Technological Challenges in the New Millennium,* September 27, 1999. AETD Information Systems Center Code 588.

3 BACKGROUND, OBJECTIVES AND SCOPE

In recent years, the need to launch a constellation of spacecraft has increased dramatically, to support complex science missions and the rapidly expanding telecommunication market. To date, however, constellation launches and operations have mostly been in the planning stage, or are experiencing difficulties operating a constellation of spacecraft.

The main difficulty and concern in the aerospace community is the lack of a strategy to effectively monitor and react to conditions reported through telemetry data from multiple spacecraft. If a traditional approach is taken, the cost will simply increase proportional to the number of spacecraft.

This architecture description describes a strategy to demonstrate the onboard automation for autonomous constellation.

4 ARCHITECTURE DESCRIPTION

This section describes the proposed system architecture. The section is divided into the operational architecture and the hardware and software architecture. The operational architecture is a brief description of the simulated system. The detail of the operational architecture is described in the CONOPs documentation. The hardware and software architecture section describes the method chosen to simulate the operational architecture.

4.1 Operational Architecture

The autonomous constellation calls for the deployment of three fully autonomous spacecraft and a ground station to support a constellation mission. Each spacecraft and ground station is linked as nodes on an existing network.

The constellation is divided into smaller systems in order to demonstrate the capabilities and verify the interconnections between the systems. The high level system is the constellation as a whole, and the lower level systems include three autonomous spacecraft, ground station and the inter-nodal connectivity system. The inter-nodal connectivity system will provide the connection between the network nodes (spacecraft to spacecraft and spacecraft to ground station).

A pictorial overview of the constellation system is shown in Figure 1.

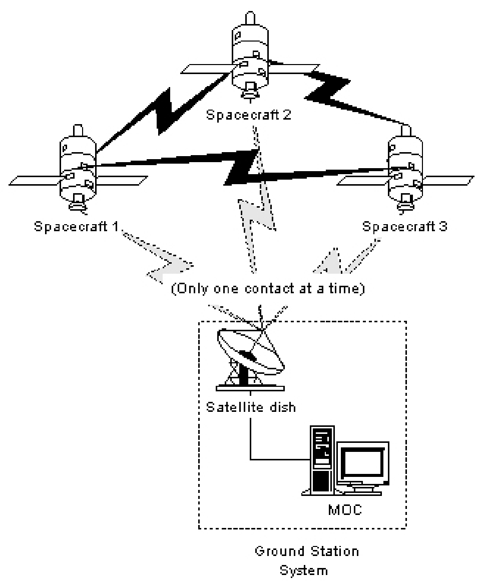


Figure 1. Constellation System Operational Environment Overview

The constellation system is the system responsible for fault detection isolation and recovery. The recovery ranges from the reconfiguration of a single spacecraft, ground station, or the reconfiguration of the constellation as a whole. Several modes of operations are identified and defined in the CONOPs documentation.

The spacecraft system is a spacecraft modeled using finite state modeling. The system will recognize only the spacecraft wide events and states. In order to automate the constellation, the spacecraft system also requires an even higher level of automation to react to the commands issued by the constellation system.

The ground station system is similar to the spacecraft system. The system will recognize only the ground station wide events and states. The spacecraft and the ground station will work as an identical node on the network, however, the resources on the ground station and the spacecraft are clearly different.

The inter-nodal connectivity system is responsible for the monitoring of the connection between the nodes. The system will detect any failure to the communication links between the nodes.

4.2 Hardware and Software Architecture

The spacecraft and ground station will be simulated using existing computers. The control system for the constellation will be hosted on a homogeneous architecture using PCs with Windows operating system (2000/NT). The control system will be able to support real time monitoring and control operations. An estimate of where on-board computing capabilities will be in 5 years has provided a goal for this prototype and future development. This project assumes a Pentium class processor running at 150Mhz with 64MB or more of available system memory will be standard on NASA spacecraft in this time frame. In this phase of development, the prototype will be tested against various hardware configurations to determine current needs. The delta between the prototype hardware requirements and the goal will provide the basis for a development plan to scale the automation technology for operation within the constraints of on-board capabilities.

All systems, spacecraft and ground station will be modeled by Altair's Finite State Modeling technology and ***Altairis MCS***™ will be scaled to operate within this environment.

Three laptops or low-power PCs will be used to simulate the spacecraft system. The computers will be connected through a router or a switch to enable computer-to-computer communication. The fourth computer, simulating the ground station will be connected to the router as well. This system is shown in Figure 2. TCP/IP or UDP/IP protocol will be used for the messaging system between the computers.

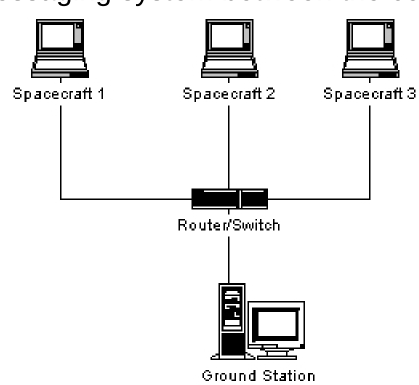


Figure 2. Hardware Architecture Setup for the Demonstration

4.2.1 Spacecraft Simulation (Laptops or low end PCs)

Altairis MCS™ is used to simulate the three spacecraft. ***Altairis MCS***™ allows several processes to run as a plug-in. In order to reduce the computational load and memory footprint, minimal plug-ins will be loaded. The following plug-ins will be loaded on the spacecraft computer.

- Messaging Plug-in: This plug-in sends and receives the state information. The plug-in also sorts, verifeis the integrity of the message and monitor the bandwidth of the communication.
- Simulator plug-in: The virtual spacecraft will be using a telemetry structure used for Wide-Field Infrared Explorer (WIRE). This plug-in sends the telemetry data, and manipulates the spacecraft data to simulate spacecraft events.
- nSource™ decomm plug-in: This nSource™ plug-in reads the telemetry data sent by the simulator plug-in. The plug-in uses a predefined WIRE data structure to decom the telemetry stream.
- nSource™ state modeling plug-in: This plug-in monitors the states of the spacecraft system. This plug-in define the states of the spacecraft system, creates transitions to autonomously control the states and also monitors for unexpected system states.
- nSource™ communication plug-in: This plug-in allows nSource™ to nSource™ communication. This plug-in uses CORBA to send messages, and will only be used for commanding the spacecraft systems.

nSource™ is the backbone of the Altair's ***Altairis MCS™***. Since the computational resource is limited (low memory, low CPU speed) there will be no Graphical User Interface (GUI) on these machines simulating the spacecraft system. However, states and status information will be stored in a file (html or xml), which can be read from remote location.

4.2.2 Ground Station Simulation (High end PC)

Altairis MCS™ is also used to simulate the ground system. Since the computational resources are not as limited as the spacecraft simulation, more plug-ins will be loaded. This machine will be designated for graphical interfaces to the spacecraft as well. Several plug-ins are identical to the plug-ins used for the spacecraft system.

- Messaging Plug-in: This plug-in sends and receives the state information. The plug-in also sorts, verifies the integrity of the message and monitor the bandwidth of the communication.
- GUI plug-in: This plug-in shows the information about the ground station system. Most of the information displayed in the graphics is obtained from the nSource™. A separate interface will be reserved to display the spacecraft messages.
- PALS Command Executor: Command line interface to the ***Altairis MCS™***.
- nSource™ decom plug-in: This nSource™ plug-in reads the telemetry data sent by the simulator plug-in. The plug-in uses a predefined WIRE data structure to decom the telemetry stream.
- nSource™ state modeling plug-in: This plug-in monitors the states of the spacecraft system. This plug-in defines the states of the ground system, creates transitions to autonomously control the states and also monitors for unexpected system states.

- nSource™ communication plug-in: This plug-in allows nSource™ to nSource™ communication. This plug-in used CORBA to send messages, and will only be used for commanding the spacecraft systems.

Figure 3 shows the summary of the software architecture superimposed onto the hardware architecture.

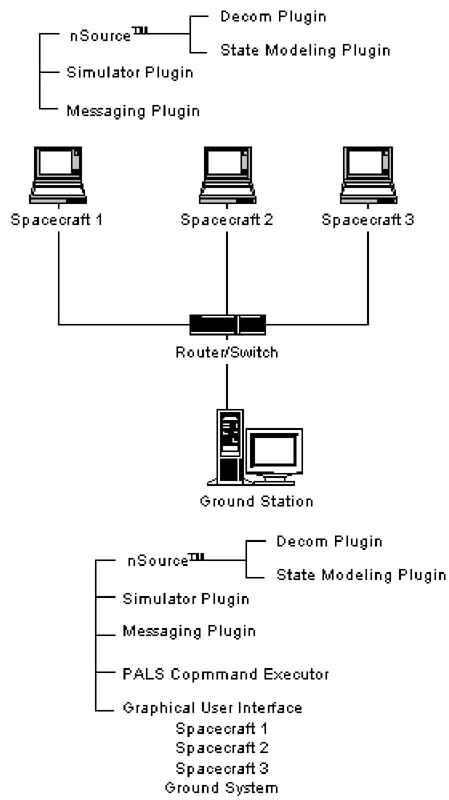


Figure 3. Hardware and Software Architecture Overview

4.3 Disadvantages and Limitations

This demonstration will only be able to demonstrate a selected subset of the domains and services available. This will be a functional demonstration based on COTS hardware for ground-based use. No space-qualified hardware will be used at this time. Follow-up projects will migrate this prototype to space qualified hardware and eventually to on-orbit test and verification.

In the absence of any real standard of spacecraft-to-spacecraft communication, the demonstration will use standard Ethernet with software limiting bandwidth to rates emulating what will be available to on orbit constellations.

4.4 Alternatives and Trade-Offs Considered

The focus of this effort is to demonstrate the functional capabilities of using the ***Altairis MCS™*** as an on-board constellation control system and to demonstrate an operations concept for autonomous constellation control.

This prototype will use the Windows2000/NT operating system as opposed to a real-time operating system. This allows for rapid development of the prototype functionality without the need for a port to a new OS. Once the prototype is functional, the system will be ported to a real-time OS in a future development phase.

An estimate of where on-board computing capabilities will be in 5 years has provided a goal for this prototype and future development. This project assumes a Pentium class processor running at 150Mhz with 64MB or more of available system memory will be standard on NASA spacecraft in this time frame. In this phase of development, the prototype will be tested against various hardware configurations to determine current needs. The delta between the prototype hardware requirements and the goal will provide the basis for a development plan to scale the automation technology for operation within the constraints of on-board capabilities.

Physical Ethernet connections will be used instead of RF connections to simulate space-to-space connections and space-to-ground connections. The space-to-space and space-to-ground visibility will also be simulated. Future development will incorporate the use of RF modems or equivalent. The Ethernet connections will be metered with software to emulated any desired rate of information flow allowing a simulation of data rates expected on-orbit.

Appendix A

Appendices are added to the document to provided easy reference to charts, classified data, etc. The location of data in appendices should be referenced in the body of the document.

Figure 4. Constellation System Operational Environment Overview.

Figure 5. Hardware Architecture Setup for the Demonstration .

Figure 6. Hardware and Software Architecture Overview .

5 NOTES AND ACRONYMS

This section contains any additional information that will aid in understanding the architecture description document. This section should include an alphabetical list of acronyms.

List of Acronyms

Term	Definition
Altair	Aeroflex Altair Cybernetics Corporation
CONOPs	Concept of Operations
GSFC	Goddard Space Flight Center
MCS	Mission Control System
MOC	Mission Operations Center
SOW	Statement of Work
TT&C	Telemetry, Tracking, and Commanding
WIRE	Wide-Field Infrared Explorer

Glossary

This section provides a clear and concise definition of terms used in this CONOPS.

Term	Definition
Finite State Model	**–** A structured approach to systems engineering which utilizes a tree-structured breakdown that includes the operational characteristics and functional relationships of the system
State	**– A**n operational state for each element in the tree-structure and is defined by functional relationships
Transitions	**–** defined steps for moving between states for each element of the structure
Configuration Manager Mode	**–** Each system only recognizes it's own states. Only one node in the network is able to recognize the state of the entire constellation and issues commands to the other spacecraft.
Conflict Mediator Mode	**–** Each system recognizes both it's own state, has knowledge of the other systems, and is able to request commands to autonomously configure the constellation. One node is responsible for receiving all the commands, selecting the most appropriate command, and issuing commands to the other spacecraft.
Synchronous Operation Mode	**–** Each system recognizes it's own states and failures. Each node is given the schedule and of the constellation and cross-nodal communication is only used to synchronize activities such tat each system is can synchronize it's own schedule.
WIRE	- A spacecraft from the Small Explorer (SMEX) Program, originally deployed to survey primarily galaxies with unusually high rates of star formation or "starburst" galaxies, which emit most of their energy in the far-infrared.